ELSEVIER
CrossMark
Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 28 (2014) 547 – 554
Procedia Computer Science

Conference on Systems Engineering Research (CSER 2014)

Eds.: Azad M. Madni, University of Southern California; Barry Boehm, University of Southern California; Michael Sievers, Jet Propulsion Laboratory; Marilee Wheaton, The Aerospace Corporation
Redondo Beach, CA, March 21-22, 2014

Increasing the probability of developing affordable systems by maximizing and adapting the solution space

Alejandro Salado[a]*, Roshanak Nilchiani[a]

[a]*Stevens Institute of Technology, Castle Point on Hudson, Hoboken 07030, USA*

Abstract

The present research suggests that the size of the solution space, which for a given set of stakeholder needs is delimited by system requirements, relates to the probability of finding affordable solutions. As a result, the effectiveness of tradespace exploration techniques is limited by its size and internal ordering. Therefore, we suggest that there exist models to elicit and use requirements that, for a given set older needs, could facilitate the maximization of the solutionof stakehspace so that the probability of finding more affordable solutions during tradespace exploration is also maximized.The present research proposes a mathematical model of the requirements elicitation process that facilitates defining performance objectives for the requirements elicitation process in striving for system affordability. The uniqueness of this research lays on two elements. First, the requirements elicitation process is mathematically modeled so that their objectives with respect to the effects on the solution space can be mathematically, and thus rigorously, described. Second, the system of interest focuses on the definition of the solutionspace as a driver for system affordability instead of on its actual exploration. The present research closes therefore the loop between stakeholder needs, system requirements, solutionspaces, and system affordability. The results of the present research are generalized to discrete requirements, fuzzy requirements, and continuous requirements or value functions.

* Corresponding author. Tel.: +49-176-32131458; fax: +1-201-216-5541.
E-mail address: asaladod@stevens.edu

doi:10.1016/j.procs.2014.03.067

Keywords: requirements; affordability; systems theory.

1. Introduction

Current worldwide economic situation stresses a global need to provide society more with less. Consequently, system affordability seems to be a concern across industry in both the commercial and the government sectors.[1,2,3] While the potential value of systems engineering is incontestable [4,5], failure in system development is still often present.[6] While it has been empirically demonstrated that requirements can strongly influence project success or failure[7,8] , an underlying theory that describes why this occurs is currently lacking. The present research explores how the size and internal ordering of the solution space influences system affordability given a set of requirements. In order to exploit the benefits of a formal systems theory [9,10] , we introduce mathematical definitions for some systems engineering activities and elements. As a result, this paper mathematically proves the effects a given set of requirements has on the probability of finding affordable solutions. We then propose a set of objectives as optimization problems and lay down some guidelines that could improve expected system affordability as the problem statement is defined.

This paper is organized as follows. Section 2 summarizes key contributions in the areas of system affordability, requirements engineering, and systems theory. Section 3 elaborates the core contribution of this paper after presenting the research approach and hypotheses. Implications of the research and limitations are discussed in section 4. The paper ends giving a short summary of the conclusions and a proposal for future work in section 5.

Nomenclature

A	system affordability
B	stakeholder benefits derived from a system performance
C	cost to develop and operate a system
CS	compliant space (part of the solution space that contains compliant or satisficing systems exist)
p_s / p_f	probability of success / probability of failure
R	set of system requirements
S	available budget during system development and operation
SN	set of stakeholder needs

2. Literature review

2.1. System affordability

Although system affordability is a desired and even needed characteristic, a widely accepted definition is lacking. However, existing research informs there are three elements that are needed to assess system affordability: cost, which refers to the expenditure that is needed to acquire and use the product or service as a function of time; value to customer, which refers to the return the paying agent has on its investment; and expenditure capability, which refers to the investment capability and will of the paying agent for a given product or service.[11,12,13,14] These three elements may be defined in monetary or non-monetary terms.

The present research assumes system affordability is a function of the aforementioned three elements.

2.2. Requirements, solution space, and impacts on affordability

Requirements, which can be expressed in the form of text, diagrams, or algorithms, can be understood as models that represent what the system is expected to achieve.[15] Therefore, they define the problem boundaries within which a solution that satisfies stakeholder needs is sought.[16] Traditionally, the system or problem boundary is defined by the result of two major activities.[16] First, the actual requirements a system has to fulfill are elicited from a set of stakeholder needs that have to be satisfied. If correctly elicited, fulfillment of requirements ensures satisfaction of stakeholder needs. Then, requirements are prioritized according to the relative preferences of the stakeholder with respect to their defined needs. Priorities enable making informed decisions under an uncertain environment.[17,18]

The influence of requirements and requirements engineering practices in project success and failure are incontestable as shown from empirical research in industrial projects.[7,8]

2.3. Systems theory

Literature suggest systems theorists and community of practice are decoupled. Whereas "the adoption of a mathematically based systems theory can provide the formulae, theorems, and proofs required to underpin the systems engineering processes and design decisions […] practical application of a systems science, particularly in terms of the use of a rigorously defined and comprehensive systems theory, is still not widespread within industry".[10] In fact, contemporary industrial surveys inform that the community of practice continues to heavily rely on heuristics and rules of thumb derived from past experiences.[19]

Systems engineering is, or at least should be, constructed on the pillars of general systems theory [20], which is defined as the "formal correspondence of general principles, irrespective of the kinds of relations or forces between the components, [which is] concerned with the principles which apply to systems in general" [21]. Under this framework, several researchers have contributed to build up a body of knowledge around the theoretical aspects of systems engineering.[22,23,24,25]

On its vast majority, such research addresses the formal description of system behavior, from the interaction of their elementary parts to their behavior as part of a given environment. In the field of requirements, for example, existing research comprehensively addresses formal definitions of requirements and their flow-down and allocation to different levels of the system decomposition.[22,26] In order to fill the void in the problem space, i.e., how stakeholder needs and system requirements relate to each other and how these affect the solution space, we proposed theoretical foundations in a previous research that related stakeholder needs, system requirements and solution spaces.[27] However, a theory that addresses the effect of the solution space on system affordability is lacking.

3. Mathematical definition of the problem statement

3.1. Approach and hypotheses

This paper mathematically proves the following two propositions:

Proposition 1. $\downarrow CS_{order_{error}} \rightarrow \uparrow p_{affordability}(t = t_n)$

Proposition 2. $\uparrow CS_{size} \rightarrow \uparrow p_{affordability}(t = t_n)$

Probability of finding an affordable solution is defined as a function of time in order to reflect the real-life time limitations in thoroughly exploring the solution space. Therefore, probability should be understood as that one to find an affordable solution within a given time.

First, the propositions are mathematically proven, which converts them into theorems of systems engineering. Then, numerical simulations are performed in order to investigate levels of actual improvement in notional cases. Finally, a set of guidelines are proposed to improve system affordability based on the resulting theorems.

It should be noted that the notion of compliant space is introduced in the propositions. This space is defined as the solution space that contains all compliant solutions to a given set of requirements.[27] The mathematical notation has been simplified with respect to the original in this paper for readability purposes, but it does not negatively impact the results presented in this paper.

3.2. Proving Proposition 1: Internal ordering of the compliant space

A set of stakeholder needs and a set of requirements can be modeled as complex numbers. Magnitude signifies the actual needs to be satisfied or requirements to be fulfilled. Phase signifies the relative priorities between needs or requirements.

$$\overline{SN_\iota} = SN_i e^{i\theta} \qquad \overline{R_\iota} = R_i e^{i\theta} \tag{1}$$

Requirements elicitation consists in defining a set of requirements given a set of stakeholder needs. Since this is a natural process, it is prone to errors, as defined in equation 2. Therefore, errors can be primarily categorized in two major types: those related to magnitude and those related to phase. The first type of errors would imply that requirements are either incorrect or incomplete, whereas the second type would imply the system is being developed under different priorities than those of the stakeholders. Therefore, the objective of any requirements elicitation technique should be to minimize the error.

$$elicit(\overline{SN_\iota}) = \widehat{R_\iota} = \overline{R_\iota} + error \tag{2}$$

In a previous work we elaborated theoretical investigations on the effects of magnitude errors on the solution space and on system affordability.[27] Therefore, this paper only addresses phase errors. First, a mathematical definition of affordability is needed. Based on the literature review presented in section 2.1, we propose equation 3. It reflects that affordability is a function of benefits, cost, available budget, and time.

$$A(t) = \begin{cases} \dfrac{k_1 B(t)}{1 + k_2 C(t)} & if\ S(t) \geq C(t) \\ 0 & if\ S(t) < C(t) \end{cases} = \left.\frac{k_1 B(t)}{1 + k_2 C(t)}\right|_{S(t) \geq C(t)} \tag{3}$$

In state of practice of systems engineering, minimization and control of phase errors are handled through requirements prioritization techniques. These are performed during the requirements elicitation process. Therefore, they are subject to the inherent assumption that priorities remain constant as the project evolves (ref. equation 4).

$$\overline{R_\iota}(t_0) = \overline{R_\iota}(t_{0+n}) \tag{4}$$

This is however in contradiction with empirical findings[28] and in particular with the definition of affordability given in equation 3. Because perceived benefits as well as available budget can evolve with time, phase errors would affect system affordability (ref. equation 5). Actual effects on system affordability are summarized in Table 1.

$$\frac{\Delta A}{\Delta \emptyset} \cong \frac{k_1 \frac{\Delta B}{\Delta \emptyset}}{1 + k_2 \frac{\Delta C}{\Delta \emptyset}} \tag{5}$$

Assuming randomness and symmetry between cases 2 and 3, it is trivial to determine that not updating requirement priorities as a system development evolves has a 0.75 probability of reducing system affordability in scenarios where priorities change with time. This result is equivalent to state the Proposition 1 is proven with a 0.75 confidence.

3.3. Proving Proposition 2: Size of the compliant space

The probability of finding an affordable solution is given by two terms (ref. equation 6). The second one is absolute and is given by the amount of affordable solutions for a given problem, relative to the amount of solutions within the universe of systems. The universal space includes all valid, non-valid, compliant, and non-compliant solutions. The first term is a modifier that represents the effectiveness of the design method against a particular solution space in seeking an acceptable solution. This term is necessary in order to reflect actual design activities, which do not randomly seek for solutions, but quickly identify a subset of the universal space where it is more probable to find an acceptable solution for a given problem.

Table 1. Case summary.

ID	$\frac{\Delta B}{\Delta \emptyset}$	$\frac{\Delta C}{\Delta \emptyset}$	$\frac{\Delta A}{\Delta \emptyset}$	Comments / Justification
1	≥ 0	N/A	N/A	Stakeholder needs and their priorities define the maximum level of satisfaction or benefits stakeholders can get. Consequently, phase errors cannot increase the benefits for the stakeholders. In fact, modification of priorities can only reduce the perceived benefits in case requirements de-scoping is needed.
2	< 0	≥ 0	< 0	Affordability would be reduced, as given by equation 5.
3	< 0	< 0	X	In this case, equation 5 does not enable determining the effect on system affordability and as a result it remains undetermined.

$$p_{affordability} = K \frac{n_{affordable}}{n_{universe}} \tag{6}$$

Since the amount of solutions in the universal space is constant, probabilities of finding affordable solutions in two compliant spaces are related as described in the first part of equation 7. If we consider valid the structure of the definition of affordability given in the previous section, a good initial assumption is that system affordability is uniformly distributed given a compliant space. In this case, the ratio between the amounts of affordable solutions in two spaces is equivalent to the ratio of their sizes. In addition, the underlying objective of proving Proposition 2 is to determine the effects of adding or removing requirements to or from a given requirement set. In this case, one of the compliant spaces is actually a subset of the other one.[27] Therefore, the resulting compliant spaces are equivalent from the design method standpoint, because they are equivalent within the universal space. Consequently, the effectiveness of design methods to quickly reach a subset of solutions that are close to the acceptable ones can be assumed to be the same. Under these terms, probabilities of finding affordable solutions in two compliant spaces become related by their sizes.

$$\begin{matrix} p_{aff}(CS_1) = K_1 \dfrac{n_{aff}(CS_1)}{n_{univ}} \\ p_{aff}(CS_2) = K_2 \dfrac{n_{aff}(CS_2)}{n_{univ}} \end{matrix} \rightarrow p_{aff}(CS_1) = p_{aff}(CS_2) \left. \frac{K_1 n_{aff}(CS_1)}{K_2 n_{aff}(CS_2)} \right|_{\substack{afford=\mathcal{U}(x,y) \\ CS_2 \subset CS_1}} \approx p_{aff}(CS_2) \frac{CS_{1_{size}}}{CS_{2_{size}}} \tag{7}$$

The second part of equation 7 proves the hypothesis described in Proposition 2. It confirms that increasing the size of the compliant space increases the probability of finding an affordable solution.

Equation 8 dictates the probability of finding an affordable solution for one try, i.e., for one design iteration. However, tradespace exploration, or more generically a design activity, consists of multiple iterations or rework cycles that are limited by available resources, i.e., solutions are sought until a satisfactory one is found or the allocated resources are exhausted. The first part of equation 8 determines the probability of finding an affordable solution within a given number of design iterations. It also reflects the fact that the probability of finding an affordable solution at each iteration is different. Such change of probability results from learning during the design process, anchoring-type biases, and reduction of amount of non-affordable solutions as they are discarded at each failed iteration. For the purpose of this paper, we suggest using the second part of equation 8, which is a simplified variant that assumes no learning, no anchoring bias, and no reduction of non-affordable solutions. The first assumption is considered acceptable as a first approach because of potential cancellation effects between learning and anchoring bias. Acceptability of the second assumption is justified by considering the amount of potential solutions significantly higher than the amount of iterations.

$$p_{aff_n} = p_{s_1} + p_{f_1}p_{s_2} + \cdots + p_{f_1}\cdots p_{f_{n-1}}p_{s_n}\big|_{TS_{size}\gg n \to p_{s_1}\approx p_{s_2}\approx\cdots\approx p_{s_n}} \approx p_s\sum_{i=0}^{n-1}(1-p_s)^i \tag{8}$$

Using therefore equation 9, sensitivity analyses on amount of iterations and relative solution space size can be performed by means of numerical analysis. Figure 1 shows probability improvements for finding affordable solutions as a function of relative solution space sizes varying from 1.10 to 1.50 and number of iterations varying from 1 to 10. Figure 1a has been obtained assuming a probability of success of 0.10, whereas a value of 0.01 was used for Figure 1b. Although the impact on maximum level of increase of probability of finding affordable solutions is visible, the biggest impact occurs for compensating with design iteration. As the probability of success decreases, the required amount of rework cycles or design iterations to compensate the effect of the compliant space size dramatically increases.

$$\frac{p_{aff_n}(CS_1)}{p_{aff_n}(CS_2)} = \frac{CS_{1_{size}}}{CS_{2_{size}}}\frac{\sum_{i=0}^{n-1}\left(1-p_s\frac{CS_{1_{size}}}{CS_{2_{size}}}\right)^i}{\sum_{i=0}^{n-1}(1-p_s)^i} \tag{9}$$

As anticipated, the probability of finding affordable solutions increases with the size of the compliant space. The difference in probability could be mitigated though by increasing the amount of rework cycles. However, as previously explained, since the availability of resources is limited, probability of finding affordable solutions needs to be defined as a function of time. As the results show, increasing the size of the compliant space dramatically reduces the required design effort to find desired solutions.

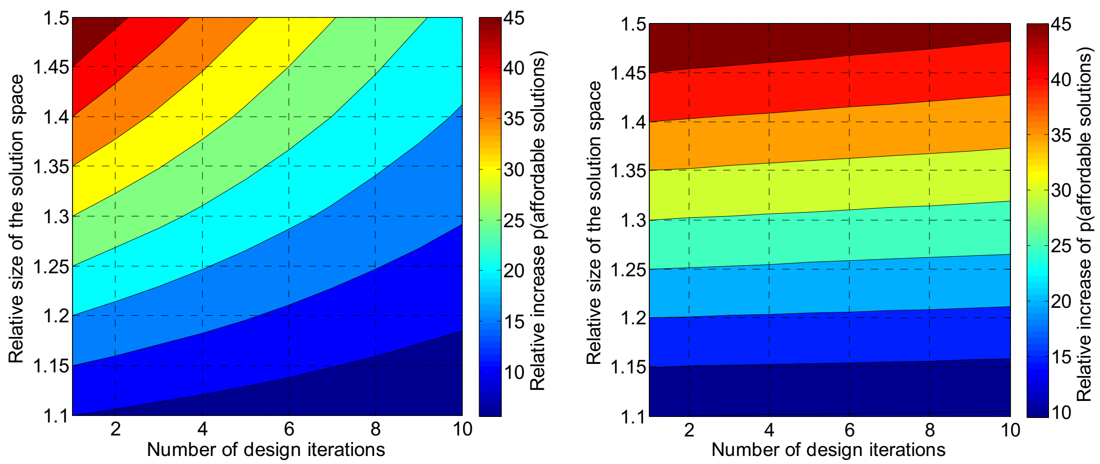

Fig. 1. (a) Relative increase probability of affordability for p_s=0.10; (b) Relative increase probability of affordability for p_s=0.01.

4. Implications and limitations

4.1. Applicability to discrete, continuous, and fuzzy requirements

This research considers a requirement is any objective a system is expected to achieve. Because all mathematical formulations are based on this premise, the results of this research are not only applicable to traditional requirements in the form of discrete targets, but also to continuous requirements in the form of value functions and to fuzzy requirements in the form of objectives without target.

4.2. Objectives for requirements engineering techniques

The confidence level achieved in proving Proposition 1 informs the need to develop evolutionary requirement prioritization techniques that are able to adapt stakeholder preferences depending on the each particular decision objective along the system life cycle.

Formal proof of Proposition 2 informs the need to develop requirement elicitation and analysis techniques that enable maximizing the size of the solution space for a given set of stakeholder needs. Using theorems that describe the effects requirements have on the solution space[27], this could be achieved by eliminating requirements that do not directly support satisfaction of stakeholder needs and detecting and decoupling conflicting requirements that are extremely difficult to be fulfilled simultaneously. Figure 1 shows a potential activity framework that would satisfy these needs.

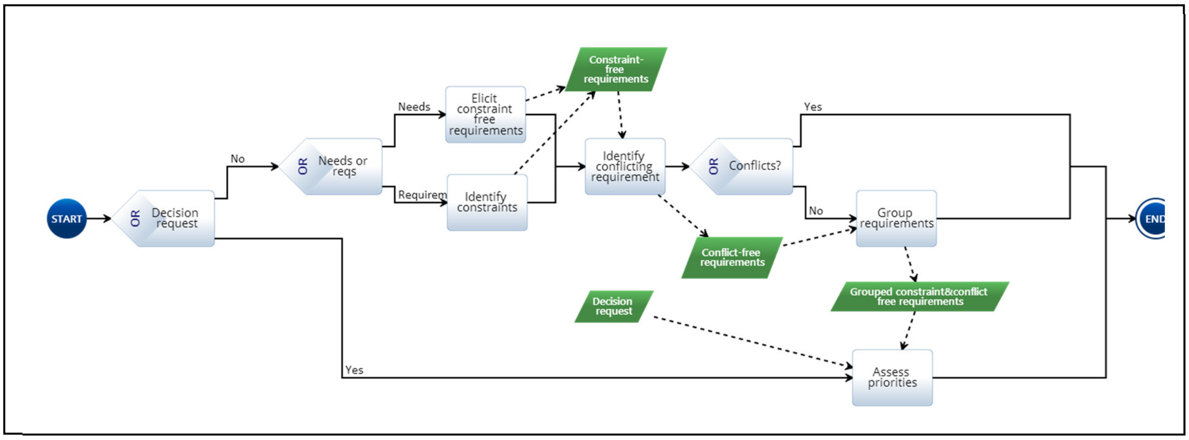

Fig. 1. Potential activity framework for maximizing and adapting the solution space

4.3. Limitations

The present research has been bounded by the following limitations, which would need to be investigated in future research:

(1) Distribution of affordable solutions is considered uniform in the universal solution space.
(2) For a given problem, the compliant space contains many more solutions than design iterations could be performed.
(3) Effects of learning and anchoring biases during the design activities are assumed to cancel each other.

5. Conclusions

This paper has mathematically proven how the size and order of the solution space affect system affordability. In particular, this paper has presented two theorems:

Theorem 1. $\downarrow CS_{order_{error}} \rightarrow \uparrow p_{affordability}(t = t_n)$.

Theorem 2. $\uparrow CS_{size} \rightarrow \uparrow p_{affordability}(t = t_n)$.

The first theorem formally justifies the need to use evolutionary requirement prioritization techniques so that defined priorities reflect stakeholder preferences at time of making decisions. The second theorem, in combination with previous research, demonstrates that increasing the amount of system requirements reduces the probability of finding affordable solutions. Consequently, it justifies the need to use the minimum set of requirements needed to satisfy a given set of stakeholder needs.

We propose and plan to continue research in this field, in particular:

1) Investigate the sensitivity of p_s on p_{aff}.
2) Investigate the sensitivity of uniformity distribution of solutions within the solution space on p_{aff}.
3) Investigate the sensitivity of the amount of solutions within the solution space on p_{aff}.
4) Investigate the effects of learning and anchoring biases in the design exploration activity.
5) Using actual project data, investigate the relation between amount of discarded or added requirements and the resulting amount of included or excluded solutions in the solution space.

Acknowledgements

This work was supported in part by the DARPA/NASA Ames Contract Number NNA11AB35C on the Fractionated Space Systems F6 project.

References

[1] Winter U. Automobil-Industrie: "Krise dauert bis 2010". aid_639188.html. _2010-bis-dauert-http://www.rponline.de/wirtschaft/news/Krise Accessed Dec 17th 2009.

[2] Government Accountability Office (GAO). *Defense Acquisitions: Assessments of Major Weapons Programs*; 20011.

[3] Bitten R, Emmons D, Bordi F, Scolese C. Explanation of Change (EoC) Study: .Approach and Findings *IEEE Aerospace Conf* 2013.

[4] Honour E. Understanding the value of systems engineering. *INCOSE Int Symp* 2004.

[5] Boehm B, Valerdi R, Honour E. The ROI of Systems Engineering: Some Quantitative Results for Software-Intensive Systems. *Syst. Eng.* 2008;**11**:221-234.

[6] Slegers NJ, Kadish RT, Payton GE, Thomas J, Griffin MD, Dumbacher D. Learning from failure in systems engineering: a panel discussion. *Syst. Eng.* 2012;**15**:74-82.

[7] Yeo KT. Critical failure factors in information system projects. *Int. J. Project Manag.* 2002;**20**:241-246.

[8] Verner J, Cox K, Bleistein S, Cerpa N. Requirements Engineering and Software Project Success: an industrial survey in Australia and the U.S. *Australasian Journal of Information Systems, North America* 13, Apr. 2007.

[9] Simpson JJ, Simpson, MJ. Formal systems concepts. *Conference on Systems Engineering Research*, 2006.

[10] Shell T. System function implementation and behavioral modeling: a systems theoretic approach. *Syst. Eng*. 2001;**4**:58-75.

[11]Xu Y, Elgh F, Erkoyuncu JA, Bankole OO, Goh Y, Cheung W, Baguley P, Wang Q, Arundachawat P, Shehab E, Newnes L, Roy R. Cost Engineering for Manufacturing: Current and Future Research. *International Journal of Computer Integrated Manufacturing* 2010:1-15.

[12] Redman Q, Stratton G. Why Affordability is a systems engineering metric. *The 20th Conference Digital Avionics Systems* 2011;**1**:4B3/1-4B3/12.

[13] Kroshl WM, Pandolfini PP. Affordability Analysis for DARPA Programs. *Johns Hopkins APL Technical Digest* 2000;**21**.

[14] Bankole OO, Roy R, Shehab E, Wardle P. Affordability assessment of Industrial product service system in the aerospace defence industry. *CIRP IPS2 Conference 2009 on Industrial Product Service Systems*, 2009.

[15] Nagasaka E. Requirements and Theories of Meaning. *International Conference on Engineering Design, ICED07* 2007.

[16] Buede DM. *The Engineering Design of Systems: Models and Methods*. 2 edition. Wiley; 2009.

[17] Sudin MN, Ahmed-Kristensen S. Change in Requirements During the Design Process. *International Conference on Engineering Design* 2011.

[18] Bischof A, Blessing L. Design for Flexibility: Making Provisions for Requirement Changes. *International Conference on Engineering Design* 2007.

[19] Sheard SA, Mostashari A. Principles of complex systems for systems engineering. *Syst Eng* 2009;**12**:295-311.

[20] Klir GJ. *An approach to general systems theory*. Van Nostrand Reinhold, New York, 1969.

[21] von Bertalanffy L. *General systems theory: Foundations, development, applications*. G. Braziller, New York, 1969.

[22] Wymore AW. *Model-based systems engineering*, CRC Press, Boca Raton, Florida, 1993.

[23] Bahill AT, Dean FF. What is systems engineering? A consensus of senior systems engineers. *INCOSE 6th Annu Int Symp*, Boston, MA, 1996, pp. 503-508.

[24] Friedman G. *Constraint theory: Multidimensional mathematical model* management. Springer, New York, 2005.

[25] Warfield JN, Hill JD. A Unified Systems Engineering Concept. *Columbus: Battelle Memorial Institute, Monograph* 1972;**1**.

[26] Thompson KR. "General system" defined for predictive technologies of A-GSBT (Axiomatic-General Systems Behavioral Theory). *Scientific Inquiry* 2006;**7**:1-11.

[27] Salado A, Nilchiani R. Aspects of a Formal Theory of Requirements Engineering: Stakeholder Needs, System Requirements, Solution Spaces, and Requirements' Qualities. *Submitted to Syst Eng*. 2013.

[28] Svetinovic D. Strategic Requirements Engineering for Complex Sustainable Systems. *Syst Eng* 2013;**16**.